AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Cancelled)

2.   (Previously Presented)   A method as recited in claim 31, further comprising reviewing configuration information to identify the defined attribute relationship associated with the view name in the location path expression.

3.   (Previously Presented)   A method as recited in claim 2, wherein reviewing configuration information further identifies a root level starting point associated with the view name.

4.   (Previously Presented) A method as recited in claim 2, wherein reviewing the configuration determines whether the client has permission to access the database based on the defined attribute relationship.

5.   (Cancelled)

6.   (Cancelled)

7.   (Previously Presented)   A method as recited in claim 31, wherein the client request is received according to the Simple Object Access Protocol (SOAP).

8.   (Previously Presented)   A method as recited in claim 31, wherein one of the at least one path element of the location path expression is a wildcard element.

9.   (Previously Presented)   A method as recited in claim 31, wherein one of the at least one path element of the location path expression indicates a search in a reversed direction of the predefined relationship.

10.   (Cancelled)

11.   (Cancelled)

12.   (Previously Presented) A method as recited in claim 31, wherein the database is a directory service database.

13.   (Cancelled).

14.   (Previously Presented)   A method as recited in claim 38, further comprising obtaining configuration information from the server defining the relationships linking attributes of the objects in the database and associated view names thereof.

15.   (Previously Presented) A method as recited in claim 14, wherein sending the request sends the request in a message to the server according to the Simple Object Access Protocol (SOAP).

16.   (Previously Presented) A method as recited in claim 38, wherein one of the at least one path element of the location path expression is a wildcard element.

17.   (Cancelled).

18.   (Previously Presented)   A computer program product as recited in claim 37, having further computer-executable instructions for reviewing configuration information to identify the defined attribute relationship associated with the view name in the location path expression.

19.   (Previously Presented)   A computer program product as in claim 18, wherein reviewing configuration information further identifies a root level starting point associated with the view name.

20.   (Previously Presented)   A computer program product as recited in claim 18, wherein reviewing the configuration determines whether the client has permission to access the database based on the predefined relationship.

21.   (Previously Presented)   A computer program product as recited in claim 37, wherein the database is a directory service database.

22-28. (Cancelled).

29.   (Previously Presented)   A method as recited in claim 38, wherein the server is a database server of the database.

30.   (Previously Presented)   A method as recited in claim 38, wherein the database is a directory service database.

31.   (Previously Presented) A method for accessing objects arranged in a hierarchy in a database, comprising:

storing objects in a database, wherein the objects each comprise corresponding attributes;

defining relationships linking different attributes of different objects in a relationship not identified by the hierarchy of the database, the relationship not being explicitly identified in the database, and not ascertainable by checking attribute names in the database, wherein defining the relationships includes creating pointers linking each object by a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein defining attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

receiving a client request for accessing a requested object in the database, wherein the request is entered in the format of a location path expression as an abbreviated XPath expression, having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

processing the client request comprising the location path expression by converting the abbreviated XPath expression to one or more database queries to locate the requested object in the database; and

returning the requested object and any other data specified in the location path expression to a client.

32.   (Previously Presented) A method as recited in claim 31, wherein the database is a database of a Web service, and wherein the location path expression is translated into a plurality of LDAP queries that are processed by the Web service to satisfy the client request and that are iteratively processed until the client request is satisfied.

33. (Previously Presented) A method as recited in claim 31, wherein the location path expression includes a plurality of objects related by the defined attribute relationship specified by the view name, and wherein each of the objects are separated by a forward slash.

34. (Previously Presented) A method as recited in claim 31, where in at least one of the defined attribute relationships includes a relationship between objects of different types that are linked by an attribute relationship.

35. (Cancelled)

36. (Previously Presented) A method as recited in claim 31, wherein the method further includes:

providing an application programming interface (API) from which applications on the client issue function calls to form the data path expression and to send the data path expression over a transport protocol to a Web service for directory access to the database.

37.   (Previously Presented) A computer program product for implementing a method for accessing objects arranged in a hierarchy in a database, the computer program product comprising:

one or more physical computer-readable media having stored thereon computer-executable instructions that, when executed by a processor, cause a computing system to perform the following:

store objects in a database, wherein the objects each comprise corresponding attributes;

define relationships linking different attributes of different objects in a relationship not identified by the hierarchy of the database, the relationship not being explicitly identified in the database, and not ascertainable by checking attribute names in the database, wherein defining the relationships includes creating pointers linking each object by a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein defining attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

receive a client request for accessing a requested object in the database, wherein the request is entered in the format of a location path expression, as an abbreviated XPath expression, having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

process the client request comprising the location path expression by converting the abbreviated XPath expression to one or more database queries to locate the requested object in the database; and

return the requested object and any other data specified in the location path expression to a client.

38.   (Previously Presented) A method for receiving objects arranged in a hierarchy in a database requested from the database, the method comprising:

connecting with a server providing access to objects stored in a database, wherein the objects each comprise corresponding attributes, and wherein relationships linking different attributes of different objects in a relationship not identified by the hierarchy of the database, the relationship not being explicitly identified in the database, and not ascertainable by checking attribute names in the database are defined by creating pointers linking each object by a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein the defined attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

forming a request for access to a requested object in the database, wherein the request is entered in the format of a location path expression as an abbreviated XPath expression, having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

sending the request comprising the location expression to cause the server to locate the requested object by converting the abbreviated XPath expression to one or more database queries to locate the requested object in the database; and

receiving the requested object and any other data specified in the location path expression.

39.   (Previously Presented) A computer program product for implementing a method for accessing objects arranged in a hierarchy in a database, the computer program product comprising:

one or more physical computer-readable media having stored thereon computer-executable instructions that, when executed by a processor, cause a computing system to perform the following:

connect with a server providing access to objects stored in a database, wherein the objects each comprise corresponding attributes, and wherein relationships linking different attributes of different objects in a relationship not identified by the hierarchy of the database, the relationship not being explicitly identified in the database, and not ascertainable by checking attribute names in the database are defined by creating pointers linking each object by a defined attribute relationship with another object, and such that the defined attribute relationships comprise linked paths between the objects, as defined by their attributes, and wherein the defined relationships comprise relationships other than parent-child relationships defined by a directory hierarchy, and wherein the defined attribute relationships for linking objects enables objects of different types to be linked by the defined attribute relationships, each attribute relationship comprising a defined name;

form a request for access to a requested object in the database, wherein the request is entered in the format of a location path expression as an abbreviated XPath expression, having the following format:

a first expression component reciting a view name, wherein the view name is a particular defined name of a particular one of the defined attribute relationships; and

at least one path element defining one of the objects related by the defined attribute relationship associated with the view name and that defines at least a portion of a linked path to the requested object;

send the request comprising the location expression to cause the server to locate the requested object by converting the abbreviated XPath expression to one or more database queries to locate the requested object in the database; and

receive the requested object and any other data specified in the location path